UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/618,277	07/11/2003	Baskaran Dharmarajan	MS1-1565US	4822

22971 7590 11/16/2007

MICROSOFT CORPORATION
ONE MICROSOFT WAY
REDMOND, WA 98052-6399

EXAMINER	
LE, MIRANDA	

ART UNIT	PAPER NUMBER
2167	

NOTIFICATION DATE	DELIVERY MODE
11/16/2007	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

roks@microsoft.com
ntovar@microsoft.com
a-rydore@microsoft.com

MN

Office Action Summary	Application No.	Applicant(s)	
	10/618,277	DHARMARAJAN ET AL.	
	Examiner	Art Unit	
	Miranda Le	2167	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *04 September 2007*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-26* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-26* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Response to Amendment

1. This communication is responsive to Amendment, filed 09/04/2007.

Claims 1-26 are pending in this application. Claims 1, 7, 13, 18, 21 are independent claims.

2. The *revised affidavit* filed on 09/04/2007 under 37 CFR 1.131 is sufficient to overcome the Ebbo reference.

Applicant's request for reconsideration of the last Office action due to Ebbo is not available as a reference under 103(a) is persuasive and, therefore, the rejection is withdrawn, and a new non-Final Action follows.

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

4. Claims 1, 3, 5, 6, 18 are rejected under 35 U.S.C. 103(a) as being unpatentable over Britton et al. (US Patent No. 6,591,289), in view of Andersen et al. (US Patent No. 6,363,398).

As to claims 1, 18, Britton teaches a method comprising:

receiving a request for a Web page *(i.e. entering the unique URL associated with the document 45a in an address bar 65a of the browser 65, col. 6, line 45 to col. 7, line 12);*

identifying an Active Server Page *(i.e. Web server 42 identifies scripted .asp-formatted files, such as scripted template file 47a, by an ".asp" file extension attached to the filename, col. 7, line 38 to col. 8, line 13)* associated with the requested Web page, wherein the Active Server Page includes a complied user interface template *(i.e. asp-formatted documents are also referred-to herein as "template" files, col. 7, line 38 to col. 8, line 13);*

executing the Active Server Page to generate the requested Web page *(i.e. the present invention utilizes scripted .asp files to retrieve information from the database server 46, to arrange the information into one of a plurality of pre-selected report layouts and to store the output therefrom as a report file formatted, preferably, in the .pdf file format, col. 8, lines 14-32); and*

providing the requested Web page to a source of the request *(i.e. report templates are stored on the file server 44 as scripted .asp-formatted files, for example, template file 147 used to generate a "Daily Status" report, and as such, each template file includes an ".asp" filename extension which signals to the file server 42 that the browser 65 is requesting that the web server 42, and more particularly, the script interpreter 48 of the web server 42, process any scripting commands found in the template file 147 before responding to the request, col. 9, lines 46-65).*

Britton teaches the Active Server Page created using a scripting language, such as Microsoft Visual Basic *(i.e. the ASP component of MSIIS permits the scripted template file 47a to include script instructions, for example, written in Microsoft Visual Basic ("VB") available*

from Microsoft Corporation, col. 6, lines 13-37; template file 147 is a scripted asp-formatted file containing script instructions written using a scripting language, such as Microsoft Visual Basic ("VB") available from Microsoft Corporation, col. 9, line 66 to col. 10, line 38).

Britton does not explicitly teach the Active Server Page created using an Active Server Page Language.

Andersen teaches the Active Server Page created using an Active Server Page Language *(i.e. the active server page can be implemented in any of the script languages supported by the Microsoft active server page facility, col. 7, lines 9-26).*

It would have been obvious to one of ordinary skill of the art having the teaching of Britton and Andersen at the time the invention was made to modify the system of Britton to include the limitations as taught by Andersen. One of ordinary skill in the art would be motivated to make this combination in order to create the active server page in view of Andersen *(col. 4, lines 17-30)*, as doing so would give the added benefit of providing a better technique for accessing database management systems that runs on server computer from within a JAVA applet running on a client computer, as taught by Andersen *(col. 3, line 51 to col. 4, line 16).*

As per claim 3, Britton teaches a method as recited in claim 1 wherein the user interface template contains HTML code *(i.e. an .html-formatted template, col. 8, line 55 to col. 9, line 19).*

As per claim 5, Britton teaches a method as recited in claim 1 wherein the Active Server Page includes a plurality of compiled user interface templates *(i.e. asp-formatted documents are also referred-to herein as "template" files, col. 7, line 38 to col. 8, line 13).*

As per claim 6, Britton teaches one or more computer-readable memories containing a computer program that is executable by a processor to perform the method recited in claim 1 *(Fig. 1).*

5. Claims 2, 7, 9, 10, 12-14, 16, 17, 19-22, 24, 25 are rejected under 35 U.S.C. 103(a) as being unpatentable over Britton et al. (US Patent No. 6,591,289), in view of Andersen et al. (US Patent No. 6,363,398), and further in view of Crater et al. (US Patent No. 5,805,442).

As to claims 7, 21, Britton teaches a method comprising:

identifying a plurality of user interface templates *(i.e. asp-formatted documents are also referred-to herein as "template" files, col. 7, line 38 to col. 8, line 13)* associated with a Web-based application *(i.e. the document 45a, col. 6, line 45 to col. 7, line 12);*.

compiling each of the plurality of user interface templates into a single file *(i.e. a new .html-formatted document 45c , col. 7, line 37 to col. 8, line 13; report templates are stored on the file server 44 as scripted .asp-formatted files, for example, template file 147 used to generate a "Daily Status" report, and as such, each template file includes an ".asp" filename extension which signals to the file server 42 that the browser 65 is requesting that the web server 42, and more particularly, the script interpreter 48 of the web server 42, process any scripting commands found in the template file 147 before responding to the request, col. 9, lines 46-65).*

Britton teaches the Active Server Page created using a scripting language, such as Microsoft Visual Basic *(i.e. the ASP component of MSIIS permits the scripted template file 47a to include script instructions, for example, written in Microsoft Visual Basic ("VB") available from Microsoft Corporation, col. 6, lines 13-37; template file 147 is a scripted asp-formatted*

file containing script instructions written using a scripting language, such as Microsoft Visual Basic ("VB") available from Microsoft Corporation, col. 9, line 66 to col. 10, line 38).

Britton does not explicitly teach:

the Active Server Page created using an Active Server Page Language;

Andersen teaches the Active Server Page created using an Active Server Page Language *(i.e. the active server page can be implemented in any of the script languages supported by the Microsoft active server page facility, col. 7, lines 9-26).*

It would have been obvious to one of ordinary skill of the art having the teaching of Britton and Andersen at the time the invention was made to modify the system of Britton to include the limitations as taught by Andersen. One of ordinary skill in the art would be motivated to make this combination in order to create the active server page in view of Andersen *(col. 4, lines 17-30)*, as doing so would give the added benefit of providing a better technique for accessing database management systems that runs on server computer from within a JAVA applet running on a client computer as taught by Andersen *(col. 3, line 51 to col. 4, line 16).*

Britton and Andersen do not specifically teach:

containing a plurality of byte codes, wherein the byte codes are capable of being executed by an execution engine;

executing the plurality of byte codes when the Web-based application is executed.

Crater teaches:

containing a plurality of byte codes, wherein the byte codes are capable of being executed by an execution engine *(i.e. ActiveX controls represent an alternative to Java, although they typically require compatible browsers and computers. These programs can be written in many*

computer languages (including Java) and usually compile to machine code, in which case they operate only in conjunction with browsers running machines with appropriate processor architectures. Some languages, however, will compile to machine-independent byte codes, which can run on a variety of processor architectures, col. 5, lines 36-44);

executing the plurality of byte codes when the Web-based application is executed *(i.e. ActiveX controls represent an alternative to Java, although they typically require compatible browsers and computers. These programs can be written in many computer languages (including Java) and usually compile to machine code, in which case they operate only in conjunction with browsers running machines with appropriate processor architectures. Some languages, however, will compile to machine-independent byte codes, which can run on a variety of processor architectures, col. 5, lines 36-44).*

It would have been obvious to one of ordinary skill of the art having the teaching of Britton, Andersen, and Crater at the time the invention was made to modify the system of Britton, Andersen to include the limitations as taught by Crater. One of ordinary skill in the art would be motivated to make this combination in order to run on variety of processor architectures in view of Crater *(col. 4, lines 36-44)*, as doing so would give the added benefit of providing the interactive capabilities made available by resources such as the World Wide Web to shift the burden of providing user interfaces for changing forms of data from monitoring computers to controllers that actually gather and report the data as taught by Crater *(col. 2, lines 40-55).*

As to claims 13, 24, Britton teaches a method comprising:

creating a plurality of user interface templates associated with a Web-based application *(i.e. a new .html-formatted document 45c , col. 7, line 37 to col. 8, line 13; report templates are stored on the file server 44 as scripted .asp-formatted files, for example, template file 147 used to generate a "Daily Status" report, and as such, each template file includes an ".asp" filename extension which signals to the file server 42 that the browser 65 is requesting that the web server 42, and more particularly, the script interpreter 48 of the web server 42, process any scripting commands found in the template file 147 before responding to the request, col. 9, lines 46-65).*

Britton teaches the plurality of user interface templates are created using a scripting language, such as Microsoft Visual Basic *(i.e. the ASP component of MSIIS permits the scripted template file 47a to include script instructions, for example, written in Microsoft Visual Basic ("VB") available from Microsoft Corporation, col. 6, lines 13-37; template file 147 is a scripted asp-formatted file containing script instructions written using a scripting language, such as Microsoft Visual Basic ("VB") available from Microsoft Corporation, col. 9, line 66 to col. 10, line 38).*

Britton does not explicitly teach:

the plurality of user interface templates are created using an Active Server Page Language.

Andersen teaches the plurality of user interface templates are created using an Active Server Page Language *(i.e. the active server page can be implemented in any of the script languages supported by the Microsoft active server page facility, col. 7, lines 9-26).*

It would have been obvious to one of ordinary skill of the art having the teaching of Britton and Andersen at the time the invention was made to modify the system of Britton to include the limitations as taught by Andersen. One of ordinary skill in the art would be motivated to make this combination in order to create the active server page in view of Andersen *(col. 4, lines 17-30)*, as doing so would give the added benefit of providing a better technique for accessing database management systems that runs on server computer from within a JAVA applet running on a client computer as taught by Andersen *(col. 3, line 51 to col. 4, line 16).*

Britton and Anderson do not specifically teach:

compiling the plurality of user interface templates into a plurality of byte codes; and

storing the plurality of byte codes associated with the plurality of user interface templates in a single file, wherein the byte codes are capable of being executed by an execution engine in a Web server.

Crater teaches:

compiling the plurality of user interface templates into a plurality of byte codes *(i.e. ActiveX controls represent an alternative to Java, although they typically require compatible browsers and computers. These programs can be written in many computer languages (including Java) and usually compile to machine code, in which case they operate only in conjunction with browsers running machines with appropriate processor architectures. Some languages, however, will compile to machine-independent byte codes, which can run on a variety of processor architectures, col. 5, lines 36-44);*

storing the plurality of byte codes associated with the plurality of user interface templates in a single file, wherein the byte codes are capable of being executed by an execution engine in a

Web server *(i.e. ActiveX controls represent an alternative to Java, although they typically require compatible browsers and computers. These programs can be written in many computer languages (including Java) and usually compile to machine code, in which case they operate only in conjunction with browsers running machines with appropriate processor architectures. Some languages, however, will compile to machine-independent byte codes, which can run on a variety of processor architectures, col. 5, lines 36-44).*

It would have been obvious to one of ordinary skill of the art having the teaching of Britton, Andersen, and Crater at the time the invention was made to modify the system of Britton, Andersen to include the limitations as taught by Crater. One of ordinary skill in the art would be motivated to make this combination in order to be able to run on variety of processor architectures in view of Crater *(col. 4, lines 36-44)*, as doing so would give the added benefit of providing the interactive capabilities made available by resources such as the World Wide Web to shift the burden of providing user interfaces for changing forms of data from monitoring computers to controllers that actually gather and report the data as taught by Crater *(col. 2, lines 40-55).*

As per claim 2, Britton, Andersen teach a method as recited in claim 1, but they do not teach wherein the user interface template has been compiled into a byte code format and the Active Server Page contains the byte codes.

Crater teaches:

the user interface template has been compiled into a byte code format and the Active Server Page contains the byte codes *(i.e. ActiveX controls represent an alternative to Java,*

although they typically require compatible browsers and computers. These programs can be written in many computer languages (including Java) and usually compile to machine code, in which case they operate only in conjunction with browsers running machines with appropriate processor architectures. Some languages, however, will compile to machine-independent byte codes, which can run on a variety of processor architectures, col. 5, lines 36-44).

It would have been obvious to one of ordinary skill of the art having the teaching of Britton, Andersen, and Crater at the time the invention was made to modify the system of Britton, Andersen to include the limitations as taught by Crater. One of ordinary skill in the art would be motivated to make this combination in order to be able to run on variety of processor architectures in view of Crater *(col. 4, lines 36-44)*, as doing so would give the added benefit of the capabilities of the Internet and, more particularly, the interactive capabilities made available by resources such as the World Wide Web to shift the burden of providing user interfaces for changing forms of data from monitoring computers to controllers that actually gather and report the data as taught by Crater *(col. 2, lines 40-55).*

As per claim 9, Crater teaches a method as recited in claim 7 wherein the plurality of byte codes are executed by an execution engine in a Web server *(Fig. 1).*

As per claim 10, Crater teaches a method as recited in claim 7 wherein the plurality of byte codes are contained in an Active Server Page *(i.e. ActiveX controls represent an alternative to Java, although they typically require compatible browsers and computers. These programs can be written in many computer languages (including Java) and usually compile to machine*

code, in which case they operate only in conjunction with browsers running machines with appropriate processor architectures. Some languages, however, will compile to machine-independent byte codes, which can run on a variety of processor architectures, col. 5, lines 36-44).

As per claim 12, Britton teaches one or more computer-readable memories containing a computer program that is executable by a processor to perform the method recited in claim 7 *(Fig. 1).*

As per claim 14, Crater teaches a method as recited in claim 13 further comprising executing the plurality of byte codes when the Web-based application is executed *(i.e. ActiveX controls represent an alternative to Java, although they typically require compatible browsers and computers. These programs can be written in many computer languages (including Java) and usually compile to machine code, in which case they operate only in conjunction with browsers running machines with appropriate processor architectures. Some languages, however, will compile to machine-independent byte codes, which can run on a variety of processor architectures, col. 5, lines 36-44).*

As per claim 16, Crater teaches a method as recited in claim 13, further comprising executing a portion of the plurality of byte codes when the Web-based application is executed *(i.e. ActiveX controls represent an alternative to Java, although they typically require compatible browsers and computers. These programs can be written in many computer*

languages (including Java) and usually compile to machine code, in which case they operate only in conjunction with browsers running machines with appropriate processor architectures. Some languages, however, will compile to machine-independent byte codes, which can run on a variety of processor architectures, col. 5, lines 36-44).

As per claim 17, Britton teaches one or more computer-readable memories containing a computer program that executable by a processor to perform the method recited in claim 13 *(Fig. 1).*

As per claim 19, Crater teaches an apparatus as recited in claim 18 wherein the Active Server Page contains a plurality of byte codes associated with the plurality of user interface templates *(i.e. ActiveX controls represent an alternative to Java, although they typically require compatible browsers and computers. These programs can be written in many computer languages (including Java) and usually compile to machine code, in which case they operate only in conjunction with browsers running machines with appropriate processor architectures. Some languages, however, will compile to machine-independent byte codes, which can run on a variety of processor architectures, col. 5, lines 36-44).*

As per claim 20, Crater teaches an apparatus as recited in claim 19 wherein the execution engine executes the byte codes associated with the request *(i.e. ActiveX controls represent an alternative to Java, although they typically require compatible browsers and computers. These programs can be written in many computer languages (including Java) and usually compile to*

machine code, in which case they operate only in conjunction with browsers running machines with appropriate processor architectures. Some languages, however, will compile to machine-independent byte codes, which can run on a variety of processor architectures, col. 5, lines 36-44).

As per claim 22, Crater teaches an apparatus as recited in claim 21 wherein the byte codes are contained in an Active Server Page *(i.e. ActiveX controls represent an alternative to Java, although they typically require compatible browsers and computers. These programs can be written in many computer languages (including Java) and usually compile to machine code, in which case they operate only in conjunction with browsers running machines with appropriate processor architectures. Some languages, however, will compile to machine-independent byte codes, which can run on a variety of processor architectures, col. 5, lines 36-44).*

As per claim 25, Crater teaches one or more computer-readable media as recited in claim 24 wherein the one or more processors further execute at least a portion of the byte codes when the Web-based application is executed *(i.e. ActiveX controls represent an alternative to Java, although they typically require compatible browsers and computers. These programs can be written in many computer languages (including Java) and usually compile to machine code, in which case they operate only in conjunction with browsers running machines with appropriate processor architectures. Some languages, however, will compile to machine-independent byte codes, which can run on a variety of processor architectures, col. 5, lines 36-44).*

6. Claims 4, 11, 23 are rejected under 35 U.S.C. 103(a) as being unpatentable over Britton et al. (US Patent No. 6,591,289), in view of Andersen et al. (US Patent No. 6,363,398), and further in view of Ponticelli et al. (US Patent No. 6,845,500).

As per claim 4, Britton, Andersen teach a method as recited in claim 1, however, they do not teach wherein the user interface template contains logic related to displaying information.

Ponticelli teaches the user interface template contains logic related to displaying information *(i.e. Behind the visible parts of the application, specifically behind its graphical user interface, lies the Application Logic Engine (ALE) which comprises the most fundamental element in the operation of this method, col. 4, lines 50-61)*

It would have been obvious to one of ordinary skill of the art having the teaching of Britton, Andersen, and Ponticelli at the time the invention was made to modify the system of Britton, Andersen to include the limitations as taught by Ponticelli. One of ordinary skill in the art would be motivated to make this combination in order to allow the creation of more flexible server/client applications in view of Ponticelli *(col. 5, line 62 to col. 6, line 2)*, as doing so would give the added benefit of providing an intelligent method that implements on-the-fly generation of client-executable or -interpretable code at the server that allows for dynamic, run-time creation of applications based on an application description repository; as taught by Ponticelli *(col. 5, lines 58-61).*

As to claims 11, 23, Britton, Andersen do not teach the byte codes include logic related to displaying information.

Ponticelli teaches the byte codes include logic related to displaying information *(i.e. Behind the visible parts of the application, specifically behind its graphical user interface, lies the Application Logic Engine (ALE) which comprises the most fundamental element in the operation of this method, col. 4, lines 50-61)*

It would have been obvious to one of ordinary skill of the art having the teaching of Britton, Andersen, and Ponticelli at the time the invention was made to modify the system of Britton, Andersen to include the limitations as taught by Ponticelli. One of ordinary skill in the art would be motivated to make this combination in order to allow the creation of more flexible server/client applications in view of Ponticelli *(col. 5, line 62 to col. 6, line 2)*, as doing so would give the added benefit of providing an intelligent method that implements on-the-fly generation of client-executable or -interpretable code at the server that allows for dynamic, run-time creation of applications based on an application description repository; as taught by Ponticelli *(col. 5, lines 58-61).*

7. Claims 8, 15, 26 are rejected under 35 U.S.C. 103(a) as being unpatentable over Britton et al. (US Patent No. 6,591,289), in view of Andersen et al. (US Patent No. 6,363,398), and Crater et al. (US Patent No. 5,805,442), and further in view of Dinovo et al. (US Patent No. 7,139,814).

As to claims 8, 15, 26, Britton, Andersen, Crater do not teach the plurality of byte codes include callback codes that call into the Web-based application code.

Dinovo teaches the plurality of codes include callback codes that call into the Web-based application code *(i.e. a page with HTML code and sends it back to the browser, col. 2, lines 35-*

49; The following pseudocode may be part of a dynamic page 182 called by the above static page 180 pseudocode to retrieve dynamic content, col. 3, lines 54-59).

It would have been obvious to one of ordinary skill of the art having the teaching of Britton, Andersen, Crater, and Dinovo at the time the invention was made to modify the system of Britton, Andersen, and Crater to include the limitations as taught by Dinovo. One of ordinary skill in the art would be motivated to make this combination in order to retrieve dynamic content that provides rich dynamic content from a database in view of Dinovo (col. 3, lines 54-59), as doing so would give the added benefit of achieving the ability for a Web site to feature dynamic content which is important in conveying information to a user, and/or attracting users to the Web siteas taught by Dinovo (col. 2, line 15-34).

Response to Arguments

8. The rejection of claims 1-12, 18-23 under 35 USC 102(e) under Ebbo, and the rejection of claims 19, 20 under 35 USC 103(a) as being unpatentable over Ebbo in view of Sisco have been withdrawn as the revised Declaration Under 37 C.F.R § 1.131 is effective to overcome the Ebbo reference (US Pub No. 20030025728).

Applicant's arguments with respect to claims 1-26 have been considered but are moot in view of the new ground(s) of rejection.

Conclusion

9. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Miranda Le whose telephone number is (571) 272-4112. The examiner can normally be reached on Monday through Friday from 8:30 AM to 5:00 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John R. Cottingham, can be reached on (571) 272-7079. The fax number to this Art Unit is (571)-273-8300.

Any inquiry of a general nature or relating to the status of this application should be directed to the Group receptionist whose telephone number is (571) 272-2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Miranda Le
November 06, 2007